

REMARKS

Claims 32, 42, 48-50, 60, 62, 64-66, 68 and 69 are all the claims pending in the application.

Initially, as explained in the Office Action, during the Examiner initiated telephone interview between Applicants' representative and the Examiner, the Examiner proposed numerous changes to claims 32, 48-50, 60 and 64-66 in order to address issues related to 35 U.S.C. 112, second paragraph, and to address prior art rejections under 35 U.S.C. 103(a). By this amendment, Applicants note that the claims have been amended so as to incorporate the majority of the Examiner's proposed changes, with the rejections and claim changes being discussed in detail below.

I. Claim Rejections under 35 U.S.C. § 112, second paragraph

Claims 32, 42, 49, 50, 60, 62, 65, 66, 68 and 69 have been rejected under 35 U.S.C. § 112, second paragraph as being indefinite.

Regarding claims 32 and 60, it is noted that the feature which included the phrase "aligned by a specific value in the source program" has been amended as proposed by the Examiner during the above-noted telephone interview so as to recite that the "directive acquisition unit detects a directive with a specific value for guaranteeing that data indicated by a pointer variable shown by the name of a specific variable in the source program is aligned by the specific value in a memory region".

Applicants respectfully submit that this modified language utilized in claims 32 and 60 addresses the Examiner's comments in the Office Action, and therefore, kindly request that the rejection be reconsidered and withdrawn.

Regarding claims 49 and 65, Applicants note that based on the Examiner's comments during the above-noted telephone interview, that the phrase "the number of development" has been replaced with the phrase --an unrolling number--, and that claims 49 and 65 have been further amended so as to set forth that the "unrolling number denotes a factor indicating how many times a code size of a loop is increased during loop unrolling." Applicants respectfully submit that this modified language utilized in claims 49 and 65 addresses the Examiner's comments in the Office Action, and therefore, kindly request that the rejection be reconsidered and withdrawn.

Regarding claims 50 and 66, the Examiner has indicated in the Office Action that each of these claims indicates that the iteration number could be an even or odd number, and that because the number zero is an even number, that claims 50 and 66 conflict with claims 48 and 64 because claims 48 and 64 indicate that the iteration value is 1 or more to eliminate the escape code. Accordingly, the Examiner has indicated that claims 50 and 66 have been interpreted to exclude the number zero. Applicants respectfully disagree with the Examiner's position for the reasons discussed below.

Initially, Applicants point out that claims 48, 50, 64 and 66 are all independent claims and, therefore, that each of these claims should each be interpreted individually, without reference to one another. In other words, Applicants submit that a feature recited in any one of

these independent claims should not affect the interpretation of another one of the other independent claims with respect to the scope thereof.

Moreover, Applicants respectfully submit that the features recited in claims 48 and 64 do not conflict with the features recited in claims 50 and 66. In particular, regarding claims 48 and 64, Applicants note that these claims are directed to the generation of an escape code being restrained when the minimum number of iterations of specific loop processing is equal to 1 or more. In contrast, Applicants note that claims 50 and 66 are not directed in any way to the generation of an escape code being restrained, but instead, are directed to the optimization unit performing optimization by loop unrolling depending on whether the guaranteed set of values is a set of only even values or a set of only odd values.

Accordingly, Applicants note that claims 48 and 64 are directed to a different aspect of the present invention than claims 50 and 66. As such, it is respectfully submitted that the features recited in claims 48 and 64 should not be used to limit the scope of claims 50 and 66. Based on the foregoing, Applicants respectfully submit that claims 48 and 64 do not conflict with claims 50 and 66, and therefore, that claims 50 and 66 should not be limited in their scope so as to exclude a value of zero.

In view of the above, Applicants respectfully submit that all of the claims satisfy the requirements of 35 U.S.C. 112, second paragraph, and therefore, kindly request that the Examiner reconsider and withdraw the above-noted rejections.

II. Claim Rejections under 35 U.S.C. § 103(a)

A. Claims 32, 60, 68 and 69 have been rejected under 35 U.S.C. § 103(a) as being unpatentable over Stallman (Using and Porting the GNU Compiler Collection for GCC 3.1)

Claim 32, as amended, recites that the directive acquisition unit detects a directive with a specific value for guaranteeing that data indicated by a pointer variable shown by the name of a specific variable in the source program is aligned by the specific value in a memory region, wherein the optimization unit performs the optimization to align the data indicated by the pointer variable in the memory region by the specific value of alignment. Regarding this claim language, Applicants note that such claim language was suggested by the Examiner during the above-noted telephone interview, and submit that Stallman does not disclose or suggest such features for at least the following reasons.

In particular, with respect to Stallman, Applicants note that this reference discloses a conventional compiler apparatus in which a directive is obtained which instructs the compiler apparatus how to compile the program. For example, regarding alignment, Stallman explains that a directive is obtained which instructs the compiler of the specific alignment that is to be used (see pages 181-182 of Stallman).

In contrast to the above-noted conventional disclosure of Stallman in which a directive for alignment instructs the compiler of the specific alignment that is to be used, according to claim 32 as amended herein, a directive is detected with a specific value for guaranteeing that data indicated by a pointer variable shown by the name of a specific variable in the source program is aligned by the specific value in a memory region, wherein the optimization unit

performs the optimization to align the data indicated by the pointer variable in the memory region specified by the specific value of alignment.

By providing the above-noted guarantee that data indicated by a pointer variable shown by the name of a specific variable in the source program is aligned by the specific value, Applicants note that the compiler apparatus according to the claimed invention is able to reduce the code size and increase execution speed.

In view of the foregoing, Applicants respectfully submit that Stallman does not disclose, suggest or otherwise render obvious at least the above-noted features recited in amended claim 32. Accordingly, Applicants submit that claim 32 is patentable over Stallman, an indication of which is kindly requested. Claims 42, 68 and 69 depend from claim 32 and are therefore considered patentable at least by virtue of their dependency.

Regarding claim 60, Applicants note that this claim recites similar features as set forth in claim 32. Accordingly, for at least similar reasons as described above, Applicants respectfully submit that the cited prior art not disclose, suggest or otherwise render obvious all of the features recited in claim 66. Accordingly, Applicants submit that claim 66 is patentable over the cited prior art, an indication of which is kindly requested. Claim 62 depends from claim 60 and is therefore considered patentable at least by virtue of its dependency.

B. Claims 48-50 and 64-66 have been rejected under 35 U.S.C. § 103(a) as being unpatentable over Stallman (Using and Porting the GNU Compiler Collection for GCC 3.1) in

view of PGI (PGI Workstation User's Guide-9 Optimization Directive and Pragmas), and further in view of Geva (US 6,539,541).

1. Claims 48 and 64

Regarding claim 48, Applicants note that this claim has been amended as proposed by the Examiner during the above-noted telephone interview in order to overcome the rejection under 35 U.S.C. 103(a), and therefore, Applicants submit that amended 48 is in condition for allowance.

In this regard, Applicants note that claim 48 has been amended so as to recite that the directive acquisition unit detects a directive with a minimum number for guaranteeing a minimum number of iterations of specific loop processing when the number of iterations of specific loop processing is not specified in the source program, wherein the optimization unit restrains generation of an escape code that is needed when the minimum number is a value equal to 1 or more.

Applicants respectfully submit that the cited prior art references do not teach or suggest at least the above-noted features of claim 48 and, therefore, respectfully submit that claim 48 is in condition for allowance, an indication of which is kindly requested.

Regarding claim 64, Applicants note that this claim recites similar features as set forth in claim 48. Accordingly, for at least similar reasons as described above, Applicants respectfully submit that the cited prior art references do not disclose, suggest or otherwise render obvious all of the features recited in claim 64. Accordingly, Applicants submit that claim 64 is patentable over the cited prior art, an indication of which is kindly requested.

2. Claims 49 and 65

Regarding claim 49, Applicants note that this claim has been amended as proposed by the Examiner during the above-noted telephone interview in order to overcome the rejection under 35 U.S.C. 103(a), and therefore, Applicants submit that amended claim 49 is in condition for allowance.

In this regard, Applicants note that claim 49 has been amended to recite the features of directive acquisition unit that detects a directive with a specific number for guaranteeing a specific number of iterations of specific loop processing in the source program, the specific number in the directive being a designated value of the number of iterations of specific loop processing when the number of iterations of specific loop processing is not specified in the source program, wherein the optimization unit performs the optimization by loop unrolling when the specific number is a value equal to or more than an unrolling number, and wherein the unrolling number denotes a factor indicating how many times a code size of a loop is increased during loop unrolling.

Applicants respectfully submit that the cited prior art references do not teach or suggest at least the above-noted features of claim 49 and, therefore, respectfully submit that claim 49 is in condition for allowance, an indication of which is kindly requested.

Regarding claim 65, Applicants note that this claim recites similar features as set forth in claim 49. Accordingly, for at least similar reasons as described above, Applicants respectfully submit that the cited prior art references do not disclose, suggest or otherwise render obvious all of the features recited in claim 65. Accordingly, Applicants submit that claim 65 is patentable

over the cited prior art, an indication of which is kindly requested.

3. Claims 50 and 66

Regarding claim 50, Applicants note that this claim has been amended so as to recite that the directive acquisition unit detects a directive for guaranteeing that the number of iterations of specific loop processing in the source program is one of a set of values, wherein the optimization unit performs the optimization by loop unrolling depending on whether the guaranteed set of values is a set of only even values or a set of only odd values.

Applicants respectfully submit that the cited prior art references do not teach or suggest at least the above-noted features of claim 50.

With respect to Stallman, Applicants note that this reference discloses a conventional compiler apparatus in which a directive is obtained which instructs the compiler apparatus how to compile the program.

Thus, while Stallman discloses the ability to instruct the compiler how to compile a program, Applicants respectfully submit that Stallman does not disclose or suggest the above-noted features recited in claim 50 which indicate that the directive acquisition unit detects a directive for guaranteeing that the number of iterations of specific loop processing in the source program is one of a set of values, wherein the optimization unit performs the optimization by loop unrolling depending on whether the guaranteed set of values is a set of only even values or a set of only odd values.

With respect to PGI, Applicants note that this reference discloses the use of a directive

which instructs a compiler whether or not to perform loop unrolling (see section 9.4, Table 9-2).

Thus, while PGI discloses the ability to instruct the compiler whether or not to perform loop unrolling, Applicants respectfully submit that PGI does not disclose or suggest the above-noted features recited in claim 50 which indicate that the directive acquisition unit detects a directive for guaranteeing that the number of iterations of specific loop processing in the source program is one of a set of values, wherein the optimization unit performs the optimization by loop unrolling depending on whether the guaranteed set of values is a set of only even values or a set of only odd values.

With respect to Geva, Applicants note that this reference discloses a compiler which reads information indicating the number of loop iterations, and then performs loop processing for the indicated number of loop iterations (see col. 9, lines 9-50).

Thus, while Geva discloses the ability to read information indicating the number of loop iterations, and to merely perform loop processing for the indicated number, Applicants respectfully submit that Geva does not disclose or suggest the above-noted features recited in claim 50 which indicate that the directive acquisition unit detects a directive for guaranteeing that the number of iterations of specific loop processing in the source program is one of a set of values, wherein the optimization unit performs the optimization by loop unrolling depending on whether the guaranteed set of values is a set of only even values or a set of only odd values.

In view of the foregoing, Applicants respectfully submit that the cited prior art references do not teach, suggest or otherwise render obvious at least the above-noted features recited in amended claim 50. Accordingly, Applicants submit that claim 50 is patentable over the cited

prior art, an indication of which is kindly requested.

Regarding claim 66, Applicants note that this claim recites similar features as set forth in claim 50. Accordingly, for at least similar reasons as described above, Applicants respectfully submit that the cited prior art references do not disclose, suggest or otherwise render obvious all of the features recited in claim 66. Accordingly, Applicants submit that claim 66 is patentable over the cited prior art, an indication of which is kindly requested.

III. Conclusion

In view of the above, reconsideration and allowance of this application are now believed to be in order, and such actions are hereby solicited. If any points remain in issue which the Examiner feels may best be resolved through a personal or telephone interview, the Examiner is kindly requested to contact the undersigned at the telephone number listed below.

Respectfully submitted,

Hajime OGAWA et al.

/Kenneth W. Fields/
By: 2009.08.12 15:34:33 -04'00'

Kenneth W. Fields
Registration No. 52,430
Attorney for Applicants

KWF/krq
Washington, D.C. 20005-1503
Telephone (202) 721-8200
Facsimile (202) 721-8250
August 12, 2009